IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application of Laurence B. Boucher, et al.		Serial No:	10/724,588
Filing Date:	November 28, 2003	Examiner:	Wen Tai Lin
Atty. Docket No:	ALA-025	GAU:	2154

For: INTELLIGENT NETWORK INTERFACE SYSTEM AND METHOD FOR ACCELERATED PROTOCOL PROCESSING

April 18, 2006

Mail Stop AF
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Pre-Appeal Brief Request for Review

Applicants hereby request review of the Final Rejection dated January 18, 2006 for the above-identified application. No amendments are being filed with this request.

Accompanying this Request is a Notice of Appeal.

The review is requested for the reasons stated on the attached sheets.

Respectfully submitted,

Mark Lauer
Reg. No. 36,578
6601 Koll Center Parkway
Suite 245
Pleasanton, CA 94566
Tel: (925) 484-9295
Fax: (925) 484-9291

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the U.S. Postal Service with sufficient postage as first class mail in an envelope addressed to: MS AF, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on April 18, 2006.

Date: 4-18-06

Mark Lauer

The Final Rejection does not present a prima facie case of anticipation or obviousness for any claim for a number of reasons, as discussed in the Request for Reconsideration of February 9, 2006. A fundamental deficiency of the Final Rejection, however, which was discussed in the Request for Reconsideration as well as in a Second Request for Reconsideration of March 15, 2006 and in an Interview Summary and Request for Reconsideration of April 11, 2006, is that the primary reference relied upon, U.S. Patent No. 6,345,302 to Bennett et al., is nonenabled.

As noted on pages 3 and 4 of the February Request for Reconsideration:

> Bennett claims to automatically send an ACK for a datagram upon verifying the checksum for the datagram.[1] But the TCP protocol specifies that sending an ACK signals to the receiver of the ACK that all the data prior to that ACK number has been successfully received by the sender of the ACK. As noted in Stevens, "TCP/IP Illustrated, Volume 1, The Protocols," which was cited in Bennett and the present application, "the acknowledgement number in the TCP header means that the sender has successfully received up through but not including that byte. There is currently no way to acknowledge selected pieces of the data stream. For example, if bytes 1-1024 are received OK, and the next segment contains bytes 2049-3072, the receiver cannot acknowledge this new segment. All it can send is an ACK with 1025 as the acknowledgement number."[2] According to Bennett's preferred embodiment, however, the "NIC 2000" would automatically send an ACK with 3073 as the acknowledgement number for the example of Stevens, assuming that the checksum for bytes 2049-3072 was valid. This would indicate to the receiver of that ACK that all data up through byte 3072 was successfully received by the sender of the ACK, even though bytes 1025-2048 were never in fact received.
>
> Because Bennett makes no provision for resending lost packets, and the sender of data would believe that no prior packets need to be resent once the sender has received the ACK that would be automatically generated according to Bennett's disclosure, Bennett's preferred embodiment would cause the loss and corruption of data. In other words, Bennett's automatic sending of an ACK upon verifying the checksum for a datagram violates both the rules and the purpose of the TCP protocol.
>
> The primary purpose of the TCP protocol is the guaranteed delivery of data. Bennett's foremost objective is also to "provide an improved method and apparatus for efficiently operating a reliable communication protocol in a computer network."[3] Yet the invention actually disclosed by Bennett would destroy the reliability and guaranteed

[1] See, e.g., column 12, lines 7-11; column 16, lines 19-26.

[2] Stevens, "TCP/IP Illustrated, Volume 1, The Protocols" (1994), page 226, lines 34-38.

[3] Summary of the Invention, column 1, line 65 – column 2, line 1.

> delivery of data, thwarting the primary purposes of TCP and Bennett. For at least this reason, Bennett does not anticipate or render obvious any of the claims of the present application. Indeed, as will be discussed more fully regarding obviousness, Bennett instead demonstrates a long-standing need for the invention defined by the present claims, and a failure of others in their approach to solving that need.

The Examiner replied to this argument by stating, in an Advisory Action dated February 28, 2006:

> TCP is associated with a re-transmittion mechanism causing any lost packet to be re-transmitted, and therefore any ACK packet sent out would trusfully reflect completion of all the prior sent packets.

Applicants responded to this in the Second Request for Reconsideration, for example on pages 2 and 3 by stating:

> Applicants are well aware of the retransmission mechanism for TCP and note that TCP retransmission depends upon the failure of the sender of data to receive an ACK within a certain time period, and that Bennett thwarts this retransmission mechanism by automatically sending ACKs despite not having received all the data.
>
> As noted in Stevens, "TCP/IP Illustrated, Volume 1, The Protocols," which was cited in Bennett and the present application:
>
> > TCP provides a reliable transport layer. One of the ways it provides reliability is for each end to acknowledge the data it receives from the other end. But data segments and acknowledgements can get lost. *TCP handles this by setting a timeout when it sends data, and if the data isn't acknowledged when the timeout expires, it retransmits the data.*[4]
>
> As noted previously, Bennett claims to automatically send an ACK for a datagram upon verifying the checksum for the datagram. But the TCP protocol specifies that sending an ACK signals to the receiver of the ACK that all the data prior to that ACK number has been successfully received by the sender of the ACK. Because Bennett sends ACKs despite not having received all the data signified by its ACKs, the timeout does not expire, and the lost data is not retransmitted.

The Examiner replied to this argument by stating, in an Advisory Action dated April 3, 2006:

> Applicant is directed to lines 3-5 at page 4 of the recent remarks

[4] Stevens, page 297, lines 2-6, emphasis added.

> quoting Comer's article: "The sender keeps a record of each packet it sends and waits for an acknowledgment before sending the next packet...". Thus, as long as the ACK packet is sent out in response to valid pakcet been received (see, e.g., Bennett: col. 12, lines 7-11), the above hand-shaking type of protocol would not yield any lost packet been incorrectly acknowledged.

Applicants responded to this in the Interview Summary and Request for Reconsideration, for example on pages 2 and 3 by stating:

> The undersigned responded that there is no teaching in Bennett of sending a single packet and then waiting for an acknowledgment to arrive for that single packet or a retransmission timeout to occur for that single packet before sending the next single packet, and so the 102 rejection is still incorrect. The undersigned also noted that such a modification of Bennett would not be obvious because it would require the other side of a TCP connection to send a single packet at a time so that the side that contained Bennett's device could perform without error, and because it would slow TCP communication to a crawl. The undersigned also notes that the Examiner's proposal contradicts Bennett's objective of "efficiently operating a reliable communication protocol," as well as eviscerating basic functions of TCP, such as the sliding window protocol.
>
> Applicants further respectfully submit that the Examiner's quotation from Comer that "The sender keeps a record of each packet it sends and waits for an acknowledgment before sending the next packet..." is taken out of context. Applicants provided the pages of Comer because in the Advisory Action of February 28, 2006, the Examiner did not seem to understand the retransmission mechanism of TCP, and that quote offered a basic building block for understanding retransmission for "*most reliable protocols*."[5] TCP, however, requires a sliding window protocol that provides a mechanism to communicate multiple packets between acknowledgments without overloading the receiver. Comer begins discussing this basic function of TCP later, on page 175, by stating:
>
>> **12.5 The Idea Behind Sliding Windows**
>>
>> Before examining the TCP stream service, we need to explore an additional concept that underlies stream transmission. The concept, known as a *sliding window*, makes stream transmission efficient...
>>
>> *A simple positive acknowledgement protocol wastes a substantial amount of network bandwidth because it must delay sending a new packet until it receives an acknowledgement for the*

[5] Comer, "Internetworking with TCP/IP," Volume 1, (1991), page 173, line 38, emphasis added.

> *previous packet.*[6]

The fact that TCP uses sliding windows and sends out multiple packets before receiving an acknowledgement for the first packet is well known to those of ordinary skill in the art and discussed, for example, on page 187 of Comer, which states:

> The TCP acknowledgement scheme is called *cumulative* because it reports how much of the stream has accumulated. Cumulative acknowledgements have both advantages and disadvantages. One advantage is that acknowledgements are both easy to generate and unambiguous. Another advantage is that lost acknowledgements do not necessarily force retransmission. A major disadvantage is that <u>the sender does not receive information about all successful transmissions, but only about a single position in the stream that has been received</u>.[7]

On the same page, Comer states:

> Because (TCP) segments travel in IP datagrams, they can be lost or delivered out of order; the receiver uses the sequence number to reorder segments.[8]

This statement also makes clear that TCP does not wait to receive an acknowledgement for each packet before sending the next packet, because the packets could in that case never arrive out of order. Even Bennett recognizes that TCP segments can be received out of order, although it fails to recognize that its automatic ACK generation would destroy the basic functions and reliability of TCP.

Stevens also notes that TCP uses a sliding window protocol to send multiple packets before waiting for an acknowledgement, as is well known to those of ordinary skill in the art, distinguishing the mechanism proposed by the Examiner by stating:

> In Chapter 15 we saw that TFTP uses a stop-and-wait protocol. The sender of a data block required an acknowledgement for that block before the next block was sent. In this chapter we'll see that TCP uses a different form of flow control called a *sliding window* protocol. It allows the sender to transmit multiple packets before it stops and waits for an acknowledgement. This leads to faster data transfer, because the sender does not have to stop and wait for an acknowledgement each time a packet is sent.[9]

[6] Comer, page 175, lines 19-32, emphasis in original.

[7] Comer, page 187, lines 18-24, italics in original, underline added.

[8] Comer, page 187, lines 6-8.

[9] Stevens, page 275, lines 3-8, emphasis in original.

In addition to the fact that Bennett is nonenabled, the Final Rejection for several reasons fails to state a prima facie case of anticipation of claim 1 or any claim that depends from claim 1, as discussed on pages 4-6 of the Request for Reconsideration.

Moreover, as noted on page 8 of the Request for Reconsideration:

> Applicants respectfully assert that the Final Rejection has failed to present a *prima facie* case of anticipation of claims 17-27 and 29-40. Instead, the Final Rejection merely refers to other claims with different limitations than claims 17-27 and 29-40, giving no real indication of why claims 17-27 and 29-40 were rejected.

Furthermore, as noted on pages 12-13 of the Request for Reconsideration, there is no incentive presented by the Final Rejection to make the modification it proposed for the two claims rejected on obviousness grounds. At best, as noted on pages 13-16 of the Request for Reconsideration, the Examiner makes bald assertions unsupported by the cited references that appear instead to come from his personal knowledge, but which are not supported by affidavits as required.

For all the above reasons, applicants respectfully assert that the Final Rejection has not stated a prima facie case of anticipation or obviousness for any claim, and should be withdrawn.